Bangla Speech-to-Text Conversion using SAPI

Shaheena Sultana
Dept. of Computer Science and Engineering
Khulna University of Engineering & Technology
Khulna, Bangladesh
shaheenaasbd@yahoo.com

M. A. H. Akhand
Dept. of Computer Science and Engineering
Khulna University of Engineering & Technology
Khulna, Bangladesh
akhand@cse.kuet.ac.bd

Prodip Kumer Das
Dept. of Computer Science and Engineering
Khulna University of Engineering & Technology
Khulna, Bangladesh
prodip_cse08@yahoo.com

M. M. Hafizur Rahman
Dept. of Computer Science
KICT, International Islamic University Malaysia
Jalan Gombak, Malaysia
hafizur@iium.edu.my

***Abstract*—Speech is the most natural form of communication and interaction between humans; whereas, text and symbols are the most common form of transaction in computer systems. Therefore, interest regarding conversion between speech and text is increasing day by day for speech oriented human-computer interaction. Microsoft Corporation developed Speech Application Program Interface (SAPI) for speech related works in its Windows operating systems that includes features for only eight languages including English. So, the aim of this study is to investigate Speech-to-Text (STT) conversion using SAPI for Bangla language. Bangla is an important language with a rich heritage; 21st February is declared as the International Mother Language day by UNESCO to respect the language martyrs for the language in Bangladesh at the year of 1952. We managed SAPI to match pronunciation from continuous Bangla speech in precompiled grammar file of SAPI and SAPI returned Bangla words in English character if matches occur. The words are then used to fetch Bangla words from database and return words in true Bangla characters and to complete the sentences. Several English words for particular Bangla word in the grammar file of SAPI is found to overcome tone variation of persons as well as pronunciation variation in language communities and shown to improve overall performance of the system. Experimental study is carried out for the technique on an article from a news paper and the recognition rate was approximately 78% on an average. Although achieved performance is promising for STT related studies, we identified several elements to improve the performance and might give better accuracy. The theme of this study will also be helpful for other languages for Speech-to-Text conversion and similar tasks.**

Keywords- Speech, Text, Human-Computer Interaction

I. INTRODUCTION

Speech is the most natural form of communication and interaction between humans. Speech is a powerful, flexible, and familiar interaction modality. Speech as conversation is the medium of choice in human relations [1]. On the other hand, text and symbols are the most common form of transaction in computer systems. Therefore, interest regarding conversion between speech and text is increasing day by day for speech oriented human-computer interaction. Between bidirectional conversions, Text-to-Speech (TTS) translation (known as speech synthesis) is straight forward and easier than Speech-to-Text (STT) conversion. The motion, manner, and pronunciation of words are the aspects of voice biometrics and give difficulty recognizing the speech to convert it to text [2]. But prospects of STT are very high in healthcare, battle management, high-performance fighter aircraft, air traffic controllers, telephony and other real life application domains [3].

A numerous differences among spoken languages demand individual systematic and scientific effort for each language. Among more than six thousand distinct languages of the world [4], most of the speech related studies are related to few languages in which English is the main. Microsoft Corporation developed Speech Application Program Interface (SAPI) for speech related works in its Windows operating systems that includes features for only eight languages including English [3]. Therefore, speech to text conversion for other languages using SAPI is a highly demandable area of study.

The aim of this study is to investigate Speech-to-Text (STT) conversion using SAPI for Bangla language. The reason for selecting Bangla is that Bangla language is far behind for scientific effort although it is the fifth most spoken language [5]. Bangla is also an important language with a rich heritage; 21st February is declared as the International Mother Language day by UNESCO to respect the language martyrs for the language in Bangladesh at the year of 1952. It is the first language of Bangladesh, West Bengal and Tripura (two states in India) and is spoken by about 245 million people around the world. It is also spoken in Malawi, Nepal, Saudi Arabia, Singapore, Australia, the UAE, UK and USA by Bangla speaking people. About one sixth population of the world is speaking in Bangla [6].

Although Bangla is an important language with a rich heritage, a systematic and scientific effort for the computerization of this language has not started yet with respect to English and other languages [1]. The last decade has been marked by a new phenomenon called globalization and it has a profound impact on different domains of life – social, political and economical. It has also experienced a significant change in the communication dynamics of the world due to the advancement of information and communication technology (ICT). The nature and function of language processing is inevitably affected by these changes. Despite the importance of English language in everyday life especially in ICT field, Bangla may have huge impact for the use of ICT and dissemination of information to Bangla speaking community [5] [6].

978-1-4673-0479-5/12/$31.00 ©2012 IEEE

Among few researches regarding Bangla, research on Speech-to-Text is very limited compared with Text-to-Speech. In Bangla, a word separation algorithm is developed for continuous Bangla speech [7]. It compares noise energy and zero crossing with speech for limited words. But, it requires conversion of analog speech signal into digital, calculation of intensity and energy of frames, save as wav file to analyze, and directly computation of autocorrelation on the waveform. Other researchers develop a technique to recognize letters, vowels and consonants. The research objective is to define the basic steps to design a full functioning recognizer [1]. It required extra circuits to recognize the letters and to generate corresponding graphs. It has to go through several steps like filtering, normalization, digitalization of signal, and calculation of reflection co-efficient. Another group of researchers works on the recognition of Bangla speech using Hidden Markov Model (HMM) [8]. This research requires noise elimination using adaptive filter and separate model for each word.

In this study, we investigated a cost effective technique of Bangla Speech-to-Text conversion; no external hardware required in the system. We used Microsoft's Speech Application Programming Interface (SAPI) that is integrated with the Windows 7 operating system (OS). SAPI dramatically reduces the code overhead required for an application to use speech recognition and Text-to-Speech, making speech technology more accessible and robust for a wide range of applications [9]. The generated grammar with the words using English pronunciation of Bangla words was loaded into the SAPI. We managed SAPI to match pronunciation from continuous Bangla speech in precompiled grammar file of SAPI and SAPI returned Bangla words in English character if matches occur. The words are then used to fetch Bangla words from database and return words in true Bangla characters and so complete the sentences.

The outline of the paper is as follows. Section II explains our approach for Bangla STT in detail with a brief explanation of SAPI and its grammar structure. Section III is for experimental studies of our technique that includes experimental methodology, setup and results for a standard document. At last, we came up with the conclusions and discussions of our proposed methodology.

II. BANGLA SPEECH-TO-TEXT (STT) USING SAPI

In our study, we have used SAPI to recognize the spoken Bangla words and converted those words into Bangla text. To make paper independent, a description of SAPI and its grammar format is included herewith.

A. Speech Application Programming Interface (SAPI)

SAPI is a middleware that provides an API and a device driver interface (DDI) for speech engines to implement. Microsoft Windows 7 operating system supplies default recognition and synthesis speech engines. In Fig. 1[10], it is seen that the speech engines are either speech recognizers or synthesizers. A speech synthesis engine (i.e., synthesizer) is instantiated locally in every application that uses it, whereas a speech recognition engine (i.e., recognizer) can be either instantiated privately or shared desktop mode. Although SAPI

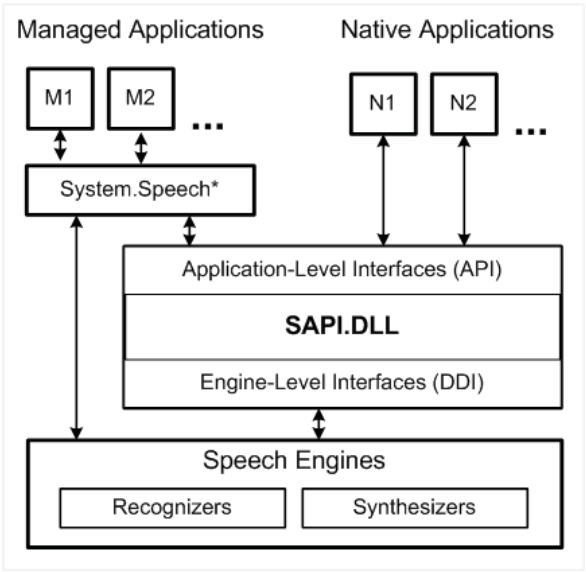

Figure 1. Working architecture of SAPI

is opaque to developers, the managed applications uses System.Speech* namespace to communicate to these engines both directly and indirectly by calling through Sapi.dll (i.e., SAPI). And native applications are used by the Windows 7 operating system [11].

Although both recognizer and synthesizers share some commonality, they can be used separately. After instantiating an engine, an application can adjust its characteristics, invoke operations on it, and register for speech event notifications [10]. Applications can choose to receive events through window messages, method callbacks, or Win32 events. These events can be filtered through the use of purposes supplied to the speech engine. For STT, speech recognition engine is involved and we managed it for our purpose.

Speech engine is language specific [10] and its recognition is processed by the grammars loaded in SAPI as XML format. XML grammars [12] are defined in the following format:

```
<GRAMMAR LANGID="409">
Grammar contents
</GRAMMAR>
```

The numeric value for attributes LANGID indicates language, here 409 for English US. The contents of an element consist of text or sub elements. Formal definitions of valid contents in this specification are provided as regular and multi-set expressions.

Grammars contents are defined in DEFINE and RULE tags. DEFINE tag contains ID name and its numeric value to define individual rule. The format is as follows:

```
<DEFINE>
    <ID NAME="RID_Sub" VAL="123"></ID>
    <ID NAME="RID_Noun" VAL="124"></ID>
</DEFINE>
```

Rule tag contains attributes like name, id, toplevel, export as follows:

<RULE NAME="BanglaSentt" ID="RID_Sub" TOPLEVEL = "ACTIVE" EXPORT="1">

Rule contents

</RULE >

Here, rule has name attribute which corresponds to the rule name. The toplevel attribute may be “ACTIVE” or “INACTIVE”. If it is “ACTIVE”, then the rule contents are checked for the recognition process. On the other hand, the rule contents are not checked for the recognition of spoken words when it is “INACTIVE” or the toplevel attribute is not present. The rule also contains sub-elements L tag to gather phrases together with PHRASE tag for words.

<L>

<PHRASE>

Contents of words

</PHRASE>

</L>

SAPI uses XML content in the following two methods [13]:

- The SAPI context-free grammar compiler compiles the XML grammar into a binary grammar format. The compiled binary grammar is loaded into the SAPI run-time environment from a file, memory, or object (.DLL) resource.
- The speech recognition [14] engine queries the run-time environment for available grammar information.

B. Bangla Speech-to-Text conversion

The goal of this study is to recognize continuous Bangla speech and extracts words from it and writes into a file; Fig. 2 shows working architecture for it. The basic steps of our methodology for Bangla STT conversion are shown in Fig. 3. Firstly, we need to generate xml grammar file based on the rules to generate xml grammar file of the previous section.

The xml grammar file is loaded after initializing SAPI. SAPI context-free grammar compiler compiles XML grammar into a binary grammar format. The compiled binary grammar is loaded into the SAPI run-time environment from a file, memory, or object (.DLL) resource [13]. Then, if a Bangla

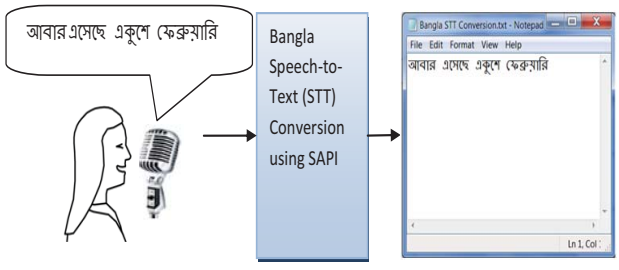

Figure 2. Working Architecture of Continuous Bangla Speech-to-Text conversion.

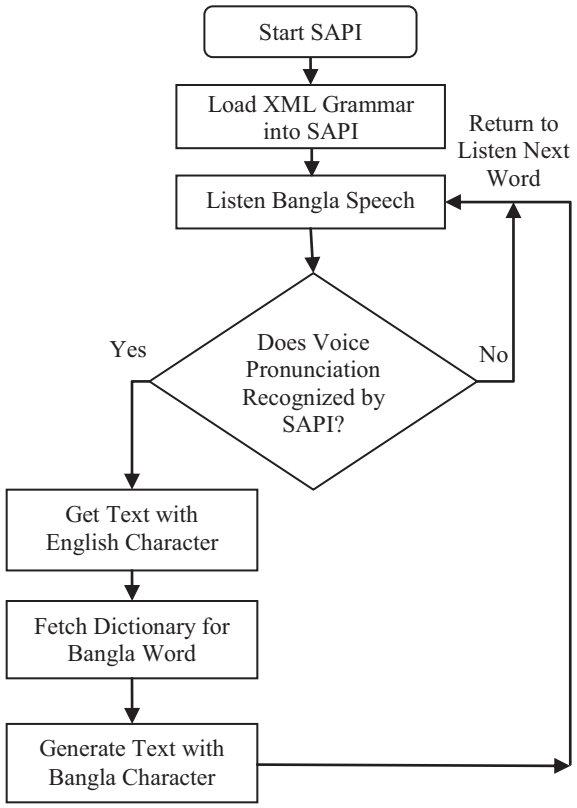

Figure 3. Bangla Speech-to-Text conversion using SAPI.

word is spoken, SAPI search it (in another word try to matches it) in the binary grammar file and returns corresponding pronunciation of Bangla word with English characters when matched occur. Our proposed methodology then go ahead with SAPI word with English character and return a word with Bangla character fetching a dictionary. We managed the dictionary with a simple database so that new word addition is simple. SAPI uses inter word gap in continuous speech [15] for word discrimination [16]. If a phrase of spoken words is misrecognized, it waits for next phrase to listen. The recognition process continues until voice phrases enters into the system.

TABLE I. PRONUNCIATION OF BANGLA WORDS WITH ENGLISH CHARACTERS WITH ONE-TO-ONE RELATIONSHIP

Bangla Word	**Pronunciation of Bangla Word with English characters**
	abar
	eseche
	ekushe
	february
	amar
	vaier
	rokte
	rangano
	din

For a better understanding, let’s take a sentence “

,

।" In this sentence, the total number of words is nine. Then we need to generate corresponding pronunciation of Bangla words with English characters, approximating the pronunciation of spoken Bangla words. Firstly, we considered nine Bangla pronunciation of words with English characters with best possible character match. Table I shows such one-to-one relationship of Bangla words and Bangla words with English characters. The xml grammar file on the bases of Table I is as follows.

```
<GRAMMAR LANGID="409">
<RULE NAME="subject" TOPLEVEL="ACTIVE">
<L>
<P>abar</P>
<P>eseche</P>
<P>ekushe</P>
<P>february</P>
<P>amar</P>
<P>vaier</P>
<P>rokte</P>
<P>rangano</P>
<P>din</P>
</L>
</RULE>
</GRAMMAR>
```

The proposed system has to recognize the words in the sentences when loaded the xml grammar into the SAPI.

We investigated the problem for xml grammar file based on one-to-one relation is that the system is highly sensitive and performance is not satisfactory. To improve the performance, we then employed all possible English characters combinations in the xml grammar file for a particular Bangla word. Table II presents additional words that is considered in one-to-many relationship with one-to-one relationship for the mentioned sentence. This approach is found to overcome tone variation of persons as well as pronunciation variation in language communities at a certain level and shown the overall performance of the system. The xml grammar file on the bases of Table 2 is as follows.

TABLE II. PRONUNCIATION OF BANGLA WORDS WITH SEVERAL POSSIBLE ENGLISH CHARACTERS COMBINATION FOR EACH WORD (ONE-TO-MANY RELATION SHIP).

Bangla Word	**Word with one-to-one relationship**	**Other possible English characters combinations for one-to-many relationship**
	abar	aabar; aber
	eseche	esece
	ekushe	ekushey; ekashe; ekuse
	february	feruary
	amar	amaar
	vaier	vaer
	rokte	rokta
	rangano	ranano; ragano

```
<GRAMMAR LANGID="409">
<RULE NAME="subject" TOPLEVEL="ACTIVE">
<L>
<P>abar</P>
<P>aabar</P>
<P>aber</P>
<P>eseche</P>
<P>esece</P>
<P>ekushe</P>
<P>ekushey</P>
<P>ekashe</P>
<P>ekuse</P>
<P>february</P>
<P>feruary</P>
<P>koma</P>
<P>kama</P>
<P>amar</P>
<P>aamaar</P>
<P>amaar</P>
<P>vaier</P>
<P>vaer</P>
<P>rokte</P>
<P>rokta</P>
<P>rokto</P>
<P>rangano</P>
<P>ranano</P>
<P>ragano</P>
<P>din</P>
<P>den</P>
</L>
</RULE>
</GRAMMAR>
```

Finally, the commands that need to be executed to load an external xml grammar files are as follows:

- To set the default dictation state[11] to inactive: grammar.DictationSetState(SpeechRuleState.SGDSInactive);
- To load the specific grammar file and set the load option to dynamic[11]: grammar.CmdLoadFromFile (@"...\grammerMultipleWordsonly.xml",SpeechLib.SpeechLoadOption.SLODynamic);
- To activate the loaded grammar to match spoken Bangla words: grammar.CmdSetRuleIdState (0, SpeechRuleState.SGDSActive).

III. EXPERIMENTAL STUDIES

This section presents experimental setup, methodology, and experimental results of our proposed method. To test the method, we processed a news article related to Bangla language from a daily newspaper.

A. Experimental Setup

To implement Bangla STT conversion, we used the following tools:

- Microsoft Visual Studio 2010
- Speech Application Programming Interface or SAPI 5.4
- Microsoft Sql Server 2008.
- Avro (Bangla writing Software)

Configuration of PC where the experiment is done is as follows:

- Processor: Intel(R) Core(TM) 2 Duo CPU E7500 @ 2.93GHz
- Memory: 2.00 GB
- System type: 32-bit Windows 7 Operating System
- One Microphone (A4TECH)
- High Definition Audio Device of Microsoft, version: 6.1.7600.16385.

B. Experimental Results

The goal of this study is continuous Bangla Speech-to-Text conversion. It requires rich xml grammar file for real life use. To recognize a speech pronunciation the corresponding word must be in xml file as well as in database; the more words the more accuracy. At this moment for testing purpose we took a Bangla newspaper article of the Daily Prothom alo dated February 21, 2011 titled " " (A National Language Planning is Required.) [17]. Fig. 4 shows a snapshot of the newspaper article. The selected article contains 396 Bangla words in total with five paragraphs including repeated words; the total distinct words were 270. We generated xml grammar file based on the article and the system is prepared to recognize speech pronunciation of the article beginning to end or from any random place.

Table III presents the experimental results paragraph wise recognition of Bangla words for both one-to-one and one-to-many relationship with pronunciation of Bangla words with English characters. From the table it observe that for first paragraph system recognize 16 words only out of 29 words when xml grammar contains a single best matched English word for a Bangla word i.e., words with one-to-one relationship. The number of word recognition increases by four and reached to 20 for the same paragraph for xml grammar file with other possible English characters combinations for a word as given example in Table II with one-to-many relationship. The reason to increase word recognition rate from 55.17% to 68.97% for multiple words for the paragraph indicates that possible combination of several English character sets for a particular Bangla is able to overcome pronunciation variations for what the technique was employed. As a specific example,

Figure 4. The document of the Daily Prothom alo that selected for testing the Bangla Speech-to-Text.

system failed to recognize the Bangla word " " with the best single English word "ekushe" and succeeded when possible English characters "ekushe", "ekushey", "ekashe", and "ekuse" are added.

From Table III it is also notable that rate of recognition is different for different paragraph. The performance of the system is worse for the first paragraph with 68.97% recognition rate and the best for third paragraph with 81.55% recognition rate. The recognition rate variation among paragraphs is reasonable because recognition is done word wise and word combinations and word repetitions are different in different paragraphs. At a glance, the overall recognition rate for the whole article was 78.54%.

TABLE III. PARAGRAPH WISE RECOGNITION OF WORDS FOR BOTH ONE-TO-ONE AND ONE-TO-MANY RELATIONSHIP OF BANGLA PRONOUACITION WITH ENGLISH CHARATERS.

Para no.	Words	one-to-one relationship		one-to-many relationship	
		Words Recognized	*Rate of Recog. (%)*	*Words Recognized*	*Rate of Recog. (%)*
1	29	16	55.17	20	68.97
2	66	42	63.64	53	80.30
3	103	70	67.96	84	81.55
4	130	88	67.69	99	76.15
5	68	48	70.59	55	80.88
Overall	**396**	**264**	**66.67**	**311**	**78.54**

TABLE IV. PARAGRAPH WISE RECOGNITION OF **DISTINCT** WORDS FOR BOTH ONE-TO-ONE AND ONE-TO-MANY RELATIONSHIP OF BANGLA PRONOUACITION WITH ENGLISH CHARATERS.

Para no.	Dist. Words	one-to-one relationship		one-to-many relationship	
		Words Recognized	*Rate of Recog. (%)*	*Words Recognized*	*Rate of Recog. (%)*
1	27	14	51.85	18	66.67
2	56	34	60.71	45	80.36
3	66	41	62.12	53	80.30
4	86	50	58.14	60	69.77
5	35	19	54.29	26	74.29
Overall	**270**	**158**	**58.52**	**202**	**74.81**

Table IV shows results for distinct Bangla words for better understanding since the article has some repeated words. It has been seen from Tables III and IV that the total distinct words in first paragraph is 27 that is close to total words (i.e., 29) and therefore recognition rate in Table IV is closed to Table III. On the other hand, total words and total distinct words difference is high for fifth paragraph, and recognition rate variation in between Table III and IV for the paragraph is notable. Finally, overall rate of recognition for the article considering distinct words is found 74.81%. The performance of investigated Bangla Speech-to-Text conversion system seems to be very good considering other speech related works.

IV. CONCLUSIONS

Speech-to-Text (STT) conversion is very prospective in various real life application domains such as healthcare, telephony systems. Microsoft Corporation provides Speech Application Program Interface (SAPI) with its Windows operating systems for speech related works for only few languages. This study investigates Speech-to-Text conversion for Bangla because Bangla language is not included in SAPI although it is an important language with a rich heritage. The idea of this study will also be applicable for other languages.

This study used English language as a middleware to manage SAPI for Bangla STT conversion. To recognize Bangla pronunciation, a xml grammar file for SAPI is generated with English character combinations for each Bangla word. The xml grammar file is then loaded into the SAPI to recognize the spoken words. SAPI returns an English character set when a spoken word is matched. Using this character set Bangla words are collected from a dictionary and write into a file. In this study, we employed several possible English character sets for a particular Bangla word that improves performance a certain level with respect to best possible English character set. The recognition rate was found 78% when the system is tested for an article of a news paper.

The important feature of the investigated Bangla STT conversion is its simplicity: as SAPI is integrated with the Microsoft Windows 7 and requires no extra software installations and hardware [16]. However, the problem of SAPI is slow and its sequential operations. In the present study, Bangla speech is recognized word by word basis. A person should speak with a proper break in each word so that system writes the word if match occurs. Parallel operation on SAPI might improve the performance and remain as future study. A faster speech recognition engine such as the Java Speech API [18] may show better performance too.

REFERENCES

[1] R. Karim, M. S. Rahman, and M. Z Iqbal, "Recognition of spoken letters in Bangla". In proc. of 5th International Conference on Computer and Information Technology (ICCIT02), Dhaka, Bangladesh, 2002.

[2] Matt Marx, M. et al., "Reliable Spelling Despite Poor Spoken Letter Recognition", in proc of the American Voice I/O Society, San Jose, California, Sep. 20-22, 1994.

[3] Wikipedia, the free encyclopedia, "Speech recognition" http://en.wikipedia.org/wiki/Speech_recognition

[4] A. de Swaan, "Words of the world: the global language system", Blackwell publishers Inc, USA, 2001.

[5] Om Gupta, "Encyclopaedia of India, Pakistan & Bangladesh (In 9 Volumes)", Isha Books, India, 2006.

[6] M. S. Islam "Research on Bangla language processing in Bangladesh: progress and challenges", in proc. of 8th International Language & Development Conference, pp. 527-533, 23-25 June 2009, Dhaka, Bangladesh.

[7] Nipa Chowdhury, Md. Abdus Sattar, Arup Kanti Bishwas "Seperating Words from Continuous Bangla Speech" Global Journal of Computer Science and Technology; vol. 9, no. 4 (2009).

[8] M. A. Hasnat, J. Mowla, and Mumit Khan, " Isolated and Continuous Bangla Speech Recognition: Implementation, Performance and application perspective", in proc. of International Symposium on Natural Language Processing (SNLP), Hanoi, Vietnam, December 2007.

[9] Microsoft Corporation, "Speech API Overview (SAPI 5.4)", http://msdn.microsoft.com/en-us/library/ee125077%28v=VS.85%29.aspx

[10] Microsoft Corporation, "Developing for Speech" http://msdn.microsoft.com/en-us/library/bb756992.aspx

[11] Microsoft Corporation, "Microsoft Speech API (SAPI) 5.3" http://msdn.microsoft.com/en-us/library/ms723627%28v=vs.85%29.aspx

[12] Microsoft Corporation, "Grammar Format Tags (SAPI 5.4)" http://msdn.microsoft.com/en-us/library/ee125671%28v=vs.85%29.aspx

[13] Microsoft Corporation, "Text grammar format overview (SAPI 5.4)" http://msdn.microsoft.com/en-us/library/ee125669%28v=vs.85%29.aspx

[14] Mark A. Fanty and Ronald A. Cole, "Spoken Letter Recognition", in proc. of Advances in Neural Information Processing Systems 3, NIPS Conference, Denver, Colorado, USA, November 26-29, 1990, pp. 220-226.

[15] K. J. Rahman, M. A. Hossain, D. Das, T. Islam, and M. G. Ali, "Continuous bangle speech recognition system, " in Proc. 6th International Conference on Computer and Information Technology (ICCIT03), Dhaka, Bangladesh, 2003.

[16] Nate Anderson, "Win 7's built-in speech recognition: a review" http://arstechnica.com/microsoft/reviews/2010/05/win-7s-built-in-speech-recognition-a-review.ars

[17] An article of The Daily Prothom alo, http://www.prothom-alo.com/detail/date/2011-02-21/news/132740

[18] Sun Microsystems, "Speech Recognition: javax.speech.recognition" http://java.sun.com/products/java-media/speech/forDevelopers/jsapi-guide/Recognition.html